10-26-00

UTILITY PATENT APPLICATION TRANSMITTAL

(Only for nonprovisional applications under 37 CFR § 1.53(b))

Attorney Docket No.	CROSS1410-1
First Inventor or Application Identifier	Michael A. Nelson, et al.
Title	System and Method for Jitter Compensation in Data Transfers
Express Mail Label No.	EL562561684US

APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents.

ADDRESS TO: Box Patent Application, Assistant Commissioner for Patents, Washington, D.C. 20231

1. [X] Fee Transmittal for FY 2000 *(Submit an original and a duplicate for fee processing)*
2. [X] Specification [Total Pages] 28 *(preferred arrangement set forth below)*
   - x - Descriptive Title of the Invention
   - ~~- Cross-References to Related Applications~~
   - ~~- Statement Regarding Fed-sponsored R & D~~
   - ~~- Reference to Microfiche Appendix~~
   - x - Background of the Invention
   - x - Brief Summary of the Invention
   - x - Brief Description of the Drawings *(if filed)*
   - x - Detailed Description
   - x - Claim(s)
   - x - Abstract of the Disclosure
3. [X] Drawing(s) *(35 USC 113)* [Total Sheets] 4
4. Oath or Declaration [Total Pages] X
   - a. [X] Newly executed (original or copy)
   - b. [ ] Copy from a prior application (37 CFR 1.63(d)) *(for continuation/divisional with Box 17 completed)*
     - i. [ ] DELETION OF INVENTOR(S) Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1.63(d)(2) and 1.33(b)
5. [ ] Incorporation By Reference *(useable if box 4b is checked)*. The entire disclosure of the prior application, from which a copy of the oath or declaration is supplied under Box 4b, is considered to be part of the disclosure of the accompanying application and is hereby incorporated by reference therein.
6. [ ] Microfiche Computer Program *(Appendix)*
7. Nucleotide and Amino Acid Sequence Submission *(if applicable, all necessary)*
   - a. [ ] Computer-Readable Copy
   - b. [ ] Paper Copy (identical to computer copy)
   - c. [ ] Statement verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

8. [X] Assignment Papers (cover sheet & document(s))
9. [ ] 37 CFR 3.73(b) Statement *(when there is an assignee)* [X] Power of Attorney
10. [ ] English Translation Document *(if applicable)*
11. [ ] Information Disclosure Statement (IDS)/PTO-1449 [ ] Copies of IDS Citations
12. [ ] Preliminary Amendment
13. [X] Return Receipt Postcard
14. [ ] Small Entity Statement(s) [ ] Statement filed in prior application, Status still proper and desired
15. [ ] Certified Copy of Priority Document(s) *(if foreign priority is claimed)*
16. [X] Other: Certificate of Express Mail; Check

17. **If a CONTINUING APPLICATION,** *check appropriate box and supply the requisite information below and in a preliminary amendment*

[ ] Continuation [ ] Divisional [ ] Continuation-In-Part (CIP) of prior Application No.: _____

Prior application information: Examiner _____ *Group / Art Unit* _____

[X] Claims the benefit of Provisional Application No. **60/202,720; Filed 05/08/00**

CORRESPONDENCE ADDRESS:

Customer Number Label:
Name: Mark L. Berrier
Ph: 512-457-7016
Fax: 512-457-7070

TYPED or PRINTED NAME Mark L. Berrier REGISTRATION NO. 35,066

SIGNATURE _____ Date 10/24/00

FEE TRANSMITTAL for FY 2000

Patent fees are subject to annual revision.
Small Entity payments must be supported by a small entity statement, otherwise large entity fees must be paid.

TOTAL AMOUNT OF PAYMENT	($) 934.00

Complete if Known	
Application Number	
Filing Date	
First Named Inventor	Michael A. Nelson, et al.
Examiner Name	
Group / Art Unit	
Attorney Docket No.	CROSS1410-1

METHOD OF PAYMENT (check one)

1. [ ] The Commissioner is hereby authorized to charge indicated fees and credit any over payments to:

Deposit Account Number: 50-0456

Deposit Account Name: Gray Cary Ware & Freidenrich LLP

[X] Charge Any Additional Fee Required Under 37 CFR §§ 1.16 and 1.17

2. [X] Payment Enclosed:

[X] Check [ ] Money Order [ ] Other

FEE CALCULATION

1. BASIC FILING FEE

Large Entity Code	$	Small Entity Code	$	Fee Description	Fee Paid
101	690	201	345	Utility Filing Fee	690
106	310	206	155	Design Filing Fee	
107	480	207	240	Plant Filing Fee	
108	690	208	345	Reissue Filing Fee	
114	150	214	75	Provisional Filing Fee	

SUBTOTAL (1) ($) 690.00

2. EXTRA CLAIM FEES

			Extra Claims X	Fee from below =	Fee Paid
Claims	27	-20	7	18.00	126
Ind. Clms	4	-3	1	78.00	78
Multiple Dependent Claims					

Large Entity Code	$	Small Entity Code	$	Fee Description
103	18	203	9	Claims in excess of 20
102	78	202	39	Indep. claims in excess of 3
104	260	204	130	Multiple dependent claim
109	78	209	39	Reissue indep. claims over original patent
110	18	210	9	Reissue claims in excess of 20 and over original patent

SUBTOTAL (2) ($) 204.00

FEE CALCULATION (continued)

3. ADDITIONAL FEES

Large Entity Code	$	Small Entity Code	$	Fee Description	Fee Paid
105	130	205	65	Surchrg – late filing fee or oath	
127	50	227	25	Surcharge – late provisional filing fee or cover sheet	
147	2520	147	2520	Filing a request for reexamination	
112	920*	112	920*	Request publication of SIR prior to Examiner action	
113	1840*	113	1840*	Request publication of SIR prior to Examiner action	
115	110	215	55	Extension for reply within first month	
116	380	216	190	Extension for reply within second month	
117	870	217	435	Extension for reply within third month	
118	1360	218	680	Extension for reply within fourth month	
119	300	219	150	Notice of Appeal	
120	300	220	150	Filing a brief in support of appeal	
121	260	221	130	Request for oral hearing	
138	1510	138	1510	Petition to institute a public use proceeding	
140	110	240	55	Petition to revive: unavoidable	
141	1210	241	605	Petition to revive: unintentional	
142	1210	242	605	Utility issue fee (or reissue)	
143	430	243	215	Design issue fee	
144	580	244	290	Plant issue fee	
122	130	122	130	Petitions to the Commissioner	
123	50	123	50	Petitions related to provisional applications	
126	240	126	240	Submission of Information Disclosure Statement	
581	40	581	40	Recording each patent assignment per property	40
146	690	246	345	Filing a submission after final rejection (37 CFR § 1.129(a))	
149	690	249	345	Each additional invention to be examined (37 CFR § 1.129(b))	
Other fee (specify)					
Other fee (specify)					

*Reduced by Basic Filing Fee Paid SUBTOTAL (3) ($) 40.00

SUBMITTED BY:				Complete (if applicable)	
Name	Mark L. Berrier, Reg. No. 35,066	Customer No.	25094	Telephone	(512) 457-7000
Signature				Date	October 24, 2000

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE	
CERTIFICATE OF MAILING BY "EXPRESS MAIL"	Atty Docket No. (Optional) **CROSS1410-1**

In the Application of: **Michael A. Nelson, et al.**
Date Filed: **October 24, 2000**
Title: **System and Method for Jitter Compensation in Data Transfers**

Attn: Box Patent Application
Hon. Asst. Commissioner of Patents
Washington, D.C. 20231

Sir:

I hereby certify that the enclosures listed below are being deposited with the United States Postal Service "EXPRESS MAIL Post Office to Addressee" service under 37 C.F.R. § 1.10, Mailing Label Certificate No. EL562561684, on October 24, 2000, addressed to Box Patent Application, Assistant Commissioner for Patents, Washington, DC 20231.

Respectfully submitted,

GRAY CARY WARE ▲ FREIDENRICH LLP

Kerry Thornhill

Enclosures:

Postcard
Check for $894.00
Utility Patent Application Transmittal
Fee Transmittal of FY 2000
Specification, 23 Claims, Abstract (28 pages)
4 Sheets of Drawings (Figures 1-4b)
Declaration and Power of Attorney
Form PTO-1595
Assignment
Check for $40.00 Assignment Recordation Fee

SYSTEM AND METHOD FOR JITTER COMPENSATION IN DATA TRANSFERS

FIELD OF THE INVENTION

The invention relates generally to data communication in computer systems, and more particularly to a system and method implemented in connection with serial data packet protocols which re-time received frames to an asynchronous transmit clock wherein overflow or underflow conditions are detected in an elastic buffer and corrected.

BACKGROUND OF THE INVENTION

Individual computer systems can be combined to form networks. There are many different types of networks, including local area networks (LANs,) wide area networks (WANs,) storage area networks (SANs) and many others. Networks are typically characterized by several characteristics, including their protocols, their architectures and their topologies.

One of the network topologies which is commonly used is a ring, or loop. This topology may be implemented, for example, in a Fibre Channel Arbitrated Loop (FC-AL.) Figure 1 shows a network which implements a loop topology.

In a network having a loop topology, data which needs to be transmitted from one device (e.g., a computer) in the network to another must be passed from the first device to the next device in the loop, which passes the data to the next device, and so on, until the data reaches the destination device. The data is typically formatted into a plurality of frames (or packets of some sort) which carry the payload (the

data) and also contain overhead information which, for example, identifies the destination device to which the data is directed. If this device is the destination of the data, the receiving device accepts the data and uses it as needed. If this device is not the destination device, the frame is transmitted to the next device in the network.

The devices in the network typically read the incoming frame data by reconstructing (from the data itself) the clock signal of the device from which the data is received. If it is determined that the data should be passed on to the next device in the loop, the data is typically retransmitted according to a clock signal which is generated within the device which is retransmitting the information. A second clock signal is used to transmit the data because, if the received clock signal were used the amount of jitter added by each node would be cumulative. As a result, the jitter could increase enough to corrupt the data.

Because the recovered receive clock and the transmit clock may have slightly different frequencies, the amount of data received in a given period may not be equal to the amount of data transmitted in that same period. As a result, the buffer in which the received data is stored until it is retransmitted may exceed its capacity (an overflow condition,) or there may not be sufficient data in the buffer to be retransmitted (an underflow condition.) This problem can be compounded each time the frames pass through one of the devices in the loop. Unless there is some means to compensate for this problem, corruption of the data may occur. A mechanism is therefore required to account for the possible differences between the clock signals in the different devices in the network.

One protocol which is used in some networks is the Fibre Channel Arbitrated Loop (FC-AL) protocol. In this protocol, several fill words are transmitted between each of the frames of data. The fill words are simply that -- non-useful or redundant words which fill in the space between frames. If the differences in the clock signals cause the frames to be transmitted more slowly than they are received (a potential overflow,) one of the fill words can be deleted from the retransmitted data in order to allow the transmission to "catch up" with the received data. If the frames are retransmitted more quickly than the data is received (a potential underflow,) a fill word can be added between frames to effectively slow down the retransmission and allow the received data to catch up.

SUMMARY

One or more of the problems outlined above may be solved by the various embodiments of the invention which, broadly speaking, comprises a system and method for inserting or deleting fill words between frames of data to compensate for frequency differences between a recovered receive clock and an internal transmit clock.

In one embodiment, the system comprises a circularly accessed buffer coupled to read and write logic. The read and write logic access data words at locations within the circular buffer as indicated by respective read and write pointers. The system further comprises control logic which compares the pointers to determine whether the buffer is approaching an underflow or overflow condition and adds or deletes fill words between frames of data to compensate for the impending underflow or overflow condition. Fill words are added or deleting by manipulating the read or write pointer to re-read or overwrite fill words when necessary. In one embodiment, the system includes fill word logic which is configured to add a fill word bit to each received word and to set or clear the fill word bit to indicate whether or not the corresponding word is a fill word. In one embodiment, the system compares read and write pointers only when one of the pointers has a value of 0 in order to reduce the number of times they are compared and thus to simplify the comparison logic.

In one embodiment, a method comprises writing words to locations in a circular buffer indicated by a write pointer, and reading words out of locations in the buffer indicated by a read pointer. The method further comprises periodically comparing the pointers to identify an approaching overflow or

underflow condition and manipulating the read or write pointers in response to the detected overflow or underflow condition. If an approaching overflow condition is detected, the write pointer is manipulated to overwrite a fillword. If an approaching underflow condition is detected, the read pointer is manipulated to re-read a fillword. In one embodiment, the method further comprises adding a fill word bit to each word stored in the buffer to facilitate identification of fill words. In one embodiment, the pointers are compared only when one of the pointers has a value of 0. In one embodiment, the fill words may be deleted with higher priority if the spacing between the read and write pointers exceeds a predetermined maximum threshold.

BRIEF DESCRIPTION OF THE DRAWINGS

Other objects and advantages of the invention may become apparent upon reading the following detailed description and upon reference to the accompanying drawings in which:

Fig. 1 is a block diagram illustrating a series of devices which are connected in a ring or loop configuration.

Figs. 2a and 2b are block diagrams illustrating the routing of data by a device (a) when the device is the destination of the data and (b) when the device is not the destination and passes the data through to a subsequent device.

Fig. 3 is a functional block diagram of the elastic buffer and associated logic in one embodiment.

Figs. 4a and 4b are very simplified state diagrams illustrating the operation of these state machines based upon the spacing of the read and write pointers in one embodiment.

While the invention is subject to various modifications and alternative forms, specific embodiments thereof are shown by way of example in the drawings and the accompanying detailed description. It should be understood, however, that the drawings and detailed description are not intended to limit the invention to the particular embodiment which is described. This disclosure is instead intended to cover all modifications, equivalents and alternatives falling within the spirit and scope of the present invention as defined by the appended claims.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

A preferred embodiment of the invention is described below. It should be noted that this embodiment and other embodiments described below are exemplary and are intended to be illustrative of the invention rather than limiting.

Broadly speaking, the invention comprises a system and method for using pointers to indicate read and write locations in a clocked circular buffer and comparing the pointers to determine whether fill words between frames should be added or deleted to prevent overflow or underflow of data from the buffer. If the write pointer leads the read pointer by more than a predetermined amount (approaching overflow,) the write pointer pauses and overwrites a fillword (effectively deleting a fillword,) allowing the read pointer to catch up to the write pointer. If the write pointer leads the read pointer by less than a predetermined amount (approaching underflow,) the read pointer pauses, re-reading the same fill word (effectively adding another fill word to the transmitted data stream) to allow the write pointer to restore the desired lead.

One embodiment of the present system is implemented in a Fibre Channel network. "Fibre Channel" refers to an data communication technology, as well as a set of standards being developed by the American National Standards Institute (ANSI) to define the technology. Fibre Channel supports both shared media and dedicated, or switched, media. Fibre Channel can be used to provide means for data transfer in many different systems, supporting workstation clusters, switched LANs, SANs (storage area networks) and the like.

Despite its name, "Fibre Channel" technology is not strictly a channel-oriented technology, nor does it represent a network topology. Fibre Channel allows devices to be interconnected in a more generalized scheme. In this scheme, devices may be connected by Fibre Channel links in configurations which include point-to-point configurations, switched fabric configurations and loop configurations. Point-to-point configurations are simply direct connections between two devices. "Switched fabric" configurations use networks formed by one or more interconnected Fibre Channel links and switches. One device in the network can be connected to any of the other devices in the network through the "fabric" formed by the links and switches. Loop configurations, such as Fibre Channel - Arbitrated Loop, or FC-AL, allow data to be communicated between any two of the devices in the network, but the data may have to pass through one or more of the other devices in the network.

Referring again to Fig. 1, the devices coupled to an FC-AL network are connected in series, with each device being coupled to receive data from a preceding device and to transmit data to a succeeding device. Referring to Figs. 2a and 2b, each of the devices 10 includes a switch 12 and an elastic buffer 14. Switch 12 is configured to route the frame, whether that frame is destined for corresponding device 10 or a subsequent device in the loop. If the data is destined for current device 10, switch 12 routes the data into the device for processing as shown in Fig. 2a. In this case, the device forwards canned fillwords to the next device in the loop. If the data is not destined for current device 10, but is instead destined for a subsequent device, switch 12 routes the data to pass through the current device as shown in Fig. 2b. If the data is passed through, it is routed to elastic

buffer 14, which is configured to transmit it to the next device in the loop.

Referring to Fig. 3, a functional block diagram of the elastic buffer and associated logic of one embodiment is shown. In this embodiment, data which is passed through a device is written by write logic 22 to buffer storage 26. Write logic 22 is clocked by the clock signal recovered from the received data. The locations to which the data is written are determined by the value of the associated write pointer 23. Write logic 22 advances write pointer 23 as each storage location is written, so that each successive half-word received from the preceding device in the loop is stored in a successive storage location in the buffer. Buffer storage 26 is configured as a circular buffer. That is, after the write pointer (or read pointer) reaches the last storage location in the buffer, it "wraps around" and points to the first storage location again.

As indicated above, one half-word is written to each storage location in this embodiment. This is a result of using 16-bit wide RAM to store data which, as defined by the Fibre Channel specification, comprises 32-bit words. It should be noted that, in other embodiments, different word lengths and storage locations of different widths may be used, so that each storage location may store full words, portions of words or multiple words. These variations are contemplated to be within the scope of this disclosure. It should be noted that, for the purposes of this disclosure, the term "word" is used to describe a portion of the data which is stored in one of the storage elements of the buffer, and is not limited to words of a particular bit-length.

In one embodiment, buffer storage 26 is implemented using dual port synchronous RAM as storage elements. Each element is 16 bits wide and can store half of a 32-bit Fibre Channel data word. The protocol defined by the Fibre Channel specification also includes a K flag in each word. The present system also includes a fill word flag in each word. This embodiment therefore includes registers to store two additional bits (the fill word bit and the K flag bit) for each of the 16-bit half-words in the RAM storage elements. Since each storage element only stores a half-word in this embodiment, the fill word bit and K flag bit for each word are stored with the corresponding high half-word.

Read logic 24 reads the data out of buffer storage 26 for transmission to the succeeding device in the loop. Read logic 24 is clocked by an internally generated transmit clock which, although it is intended to operate at the same frequency as the recovered clock signal, is independent of this signal and may therefore vary from it to some extent. Read logic 24 is configured to retrieve the data from the locations indicated by read pointer 25. In operation, read pointer 25 should point to a storage location which differs from the target of the write pointer by an amount which is implementation dependent. The Fibre Channel specification requires that the latency of data which is passed through a device to be a maximum of six word-times. The amount by which the write pointer leads the read pointer adds to the latency, so the spacing of the pointers should be minimized, but should still allow for the addition or deletion of words without corrupting data. In one embodiment, buffer storage 26 has eight storage elements, and the ideal spacing for the pointers is three or four storage locations.

The present system also includes control logic 28, which is coupled to both read logic 24 and write logic 22. Control logic 28 is configured to periodically access read pointer 25 and write pointer 23 to determine the spacing of the storage locations to which they point. If the indicated read and write locations are optimally spaced, the read and write logic proceed without interruption or interference from the control logic. If the indicated read and write locations are too close together (i.e., the write pointer leads the read pointer by less than a predetermined amount,) control logic 28 is configured to cause the read logic to re-read one of the storage locations before continuing and reading the succeeding storage locations. This causes the spacing between the pointers to increase. If, on the other hand, the indicated read and write locations are too far apart (i.e., the write pointer leads the read pointer by more than a predetermined amount,) control logic 28 is configured to cause the write logic to overwrite the same storage location. This causes the spacing between the pointers to decrease.

It should be noted that control logic 28 does not cause read logic 24 to re-read or skip words that contain any of the useful information being transmitted from the source computer to the destination computer. Doing so would corrupt the data. Only fill words, which contain redundant information, are allowed to be skipped or re-read. Thus, fill words are effectively added or deleted between frames when the data is retransmitted by the device.

In the Fibre Channel specification, specific bit sequences are defined as fill words. Fill words are normally detected by scanning each received word and determining which of the words match the sequences defined in the Fibre Channel

specification. To facilitate identification of fill words in the present system, this function is performed by fill word logic 30. Fill word logic 30 is positioned upstream from write logic 22 and is configured to add a fill word flag bit to each of the words which is written to buffer storage 26. This bit is set for the fill words and cleared for all other words. After the fill word bit is added to each of the words, fill words can be identified by checking this bit, rather than scanning the entire word to determine whether it matches one of the predefined fill word sequences.

Control logic 28 uses two state machines to control the addition or deletion of fillwords between frames. Referring to Figs. 4a and b, two state diagrams illustrating the operation of these state machines are shown. In one embodiment, the system is only periodically checked to determine whether the read and write pointers are properly spaced. Because the difference between the recovered receive clock and the internal transmit clock is expected to be minimal, there is no need to check the spacing of the pointers every time the system writes to or reads from one of the storage locations. The timing of the periodic checks is discussed in more detail below.

Before describing the operation of the state machines in control logic 28, it should be noted that the spacing between pointers in this embodiment is determined by subtracting the location of the read pointer from the location of the write pointer. In other embodiments, the spacing can be determined by subtracting the location of the write pointer from the location of the read pointer. When either state machine is considered, one of these cases will involve a comparison of the pointer spacing to a minimum threshold, while the other

case will involve a comparison of the pointer spacing to a maximum threshold. For instance, if a circularly accessed buffer has 8 storage elements, and the write pointer should lead the read pointer by no more than 5 storage locations, the spacing determined by subtracting the read pointer from the write pointer is compared to a maximum value of 5. If, on the other hand, the spacing is determined by subtracting the write pointer from the read pointer, the spacing is compared to a minimum value of 3 (8-5).

As noted above, this embodiment determines pointer spacing by subtracting the location of the read pointer from the location of the write pointer. Referring to Fig. 4a, when the spacing between pointers is checked, the addition state machine (i.e., the state machine that controls the addition of fill words) compares the pointer spacing to a predetermined minimum value. If the spacing is greater than the predetermined minimum value, it is not necessary to add a fill word, and the state machine remains in the idle state. If, on the other hand the spacing is less than the predetermined minimum value, a fill word should be added, and the state machine moves to state S1. When the state machine moves to state S1, the system examines the words which are being read from buffer storage 26 by read logic 24. As explained above, only fill words are re-read, so the control logic takes no further action until a fill word is detected. When a fill word is detected, the state machine moves to state S2, and a fill word is added to the stream of transmitted data by reading the fill word twice (the read pointer is not incremented after the first read.)

Referring to figure 4b, the operation of the deletion state machine is similar to that of the addition state

machine, except that the pointer spacing is compared to a predetermined maximum value, and a fill word is deleted (overwritten) rather than being re-read. The deletion of the fillword occurs according to the high- and low- priority conditions described in more detail below.

In an embodiment using eight storage locations, the write pointer initially leads the read pointer by 3 or 4 storage locations. If the spacing is determined by subtracting the read pointer from the write pointer, there is an initial spacing of four or five storage locations. This spacing is considered to be safe, and no action needs to be taken if this spacing is detected. If the spacing is 0, 1 or 7, the pointers are too far apart and an error is considered to have occurred. If the spacing is 2 or 3, a fill word should be added and, if the spacing is 6, a fillword should be deleted.

It should be noted that, in this embodiment, spacings of 4 or 5 are both quiescent because each storage location stores a half word, while full words are added or deleted. If only one spacing was quiescent, then each time a word was added or deleted, the pointers would end up with spacing that would indicate an opposing action. In other words, every time a word was added, the resulting spacing would indicate that a word should be deleted, and vice versa. Consequently, the system would constantly be adding and deleting words.

In one embodiment, the system may take into account the urgency of the need to add or delete a fill word. It is helpful to consider the use of fill words between frames in a Fibre Channel system. According to the Fibre Channel specification, the number of fill words between frames should be a minimum of 6 when the frames are originated and should

always be at least 2 when received. Consequently, the control logic is configured so that, if it has detected the need to delete a fill word, it will count the number of fill words between frames before deleting a fill word. If deleting one of the existing fill words would result in less than 4 fill words between the frames, the control logic may delay the deletion unless it is a high priority action. In other words, if the data may otherwise be corrupted, it may be necessary to immediately delete a fill word. If the deletion is a low priority, however, the control logic may wait until the next series of fill words is detected before deleting a fill word. (In the system described above, where pointer spacing of 2 or 3 storage locations indicate the need to delete a fill word, the spacing of 2 may be a high priority deletion, whereas a spacing of 3 may be a low priority deletion.)

In one embodiment, control logic 28 performs an asynchronous comparison between the read and write pointers and has an error of plus or minus one storage location. Even a slight variation between the receive and transmit clocks can cause this circuit to dither between two values. This may cause the control logic to add a fill word and then almost immediately delete a fill word, or vice versa. This embodiment therefore implements a form of hysteresis by including logic to suppress any addition or deletion of fill words unless the pointer spacing is stable for at least half of a clock cycle.

This embodiment also implements a dead time to the add and delete state machines. Because the read and write pointers are only checked every eighth clock cycle, it takes sometime before the spacing between the pointers can be updated to reflect the results of adding or deleting a word.

Consequently, after a fill word is added or deleted, the state machines are disabled for about 15 clock cycles so that incorrect spacing information is not generated.

In order to simplify the calculation of the spacing between the pointers, the spacing is determined when one of the pointers (more specifically, the pointer which is subtracted from the other) is at location 0. (It is assumed here that the storage locations are numbered 0 through n.) When this pointer is at 0, the value of the other pointer is equal to the distance between the pointers. Thus, rather than actually having to calculate the spacing between the pointers, it is only necessary to read the value of one of the pointers. The determination of the pointer spacing when one of the pointers is at 0 also provides a convenient timing mechanism. In other words, the pointer spacing is determined whenever the pointer is set to 0. If there are eight storage locations in the buffer, the spacing is determined every eighth read or every eighth write to the buffer. Because the difference between the transmit and receive clocks is small, the spacing will change very slowly, so checking it only periodically is sufficient.

While the present invention has been described with reference to particular embodiments, it should be understood that the embodiments are illustrative and that the scope of the invention is not limited to these embodiments. Many variations, modifications, additions and improvements to the embodiments described above are possible. It is contemplated that these variations, modifications, additions and improvements fall within the scope of the invention as detailed within the following claims.

CLAIMS:

What is claimed is:

1. A method for buffering data in a device configured to be coupled to other devices in a loop network topology, the method comprising:

storing a plurality of successive words into storage locations in a circular memory;

reading the plurality of successive words out of storage locations in the circular memory;

maintaining a read pointer indicating a read location in the circular memory;

maintaining a write pointer indicating a write location in the circular memory;

comparing the read and write pointers to determine a number of storage locations by which the write pointer leads the read pointer;

reading one of the storage locations twice in response to detecting that the number of storage locations by which the write pointer leads the read pointer is less than a predetermined minimum number; and

overwriting one of the storage locations in response to detecting that the number of storage locations by which the write pointer leads the read pointer is greater than a predetermined maximum number.

2. The method of claim 1 wherein comparing the read and write pointers to determine a number of storage locations by which the write pointer leads the read pointer is not performed each time one of the storage locations is read.

3. The method of claim 2 wherein comparing the read and write pointers to determine a number of storage locations by which the write pointer leads the read pointer comprises reading the value of one of the read and write pointers when the other of the read and write pointers has a value of 0.

4. The method of claim 1 further comprising
examining a plurality of words in the storage locations,
detecting that one of the plurality of words is a fill word, wherein reading one of the storage locations twice comprises reading the storage location containing the fill word twice, and wherein overwriting one of the storage locations comprises examining the plurality of words in the storage locations,
detecting that one of the plurality of words is a fill word and
overwriting the storage location containing the fill word.

5. The method of claim 4 further comprising determining which of the plurality of successive words are fill words and for each of the plurality of successive words providing an indication of whether the word is a fill word.

6. The method of claim 5 wherein providing the indication comprises adding a bit to each word, setting the bit if the word is a fill word and clearing the bit if the word is not a fill word.

7. The method of claim 4 further comprising:
determining whether the number of storage locations by which the write pointer leads the read pointer is

greater than a high-priority threshold number which is greater than the predetermined maximum number; and

if the number of storage locations by which the write pointer leads the read pointer is greater than the predetermined maximum number but less than the high-priority threshold number, deleting one of the fill words with a low priority, and

if the number of storage locations by which the write pointer leads the read pointer is greater than the high-priority threshold number, deleting one of fill words with a high priority.

8. The method of claim 4 further comprising:

determining whether the number of storage locations by which the write pointer leads the read pointer is greater than a high-priority threshold number which is greater than the predetermined maximum number; and

if the number of storage locations by which the write pointer leads the read pointer is greater than the predetermined maximum number but less than the high-priority threshold number, waiting until a series of more than four consecutive fill words is detected and deleting one of the series of fill words, and

if the number of storage locations by which the write pointer leads the read pointer is greater than the high-priority threshold number, waiting until a series of more than two consecutive fill words is detected and deleting one of the series of fill words.

9. The method of claim 1 wherein if the number of storage locations by which the write pointer leads the read pointer is greater than a high error threshold, an error is indicated and if the number of storage locations by which the write pointer leads the read pointer is less than a low error threshold, an error is indicated.

10. The method of claim 1 wherein comparing the read and write pointers to determine a number of storage locations by which the write pointer leads the read pointer comprises determining a value for the number of storage locations by which the write pointer leads the read pointer which is maintained for at least one half of a clock cycle.

11. The method of claim 1 wherein comparing the read and write pointers to determine the number of storage locations by which the write pointer leads the read pointer is delayed for a predetermined period of time after reading or overwriting one of the storage locations in response to detecting the spacing of the read pointer and the write pointer.

12. A method for buffering data in a device configured to be coupled to other devices in a loop network topology, the method comprising:

writing a plurality of successive words into storage locations in a circular memory indicated by a write pointer;

reading the plurality of successive words out of storage locations in the circular memory indicated by a read pointer;

determining a number of storage locations by which the write pointer leads the read pointer;

reading one of the storage locations which contains a fill word two or more consecutive times in response to detecting that the number of storage locations by which the write pointer leads the read pointer is less than a predetermined minimum number; and

overwriting one of the storage locations which contains a fill word one or more times in response to detecting that the number of storage locations by which the write pointer leads the read pointer is greater than a predetermined maximum number.

13. A buffer system comprising:

a circular buffer configured to store data in a plurality of storage locations, wherein a write position is indicated in the buffer by a write pointer and a read position is indicated in the buffer by a read pointer;

write logic configured to write received data to the storage location indicated by the write pointer;

read logic configured to read data from the storage location indicated by the read pointer; and

control logic configured to compare the positions of the read and write pointers and configured to control the read logic to adjust at least one of the read and write pointers in response to the relative positions of the read and write pointers.

14. The buffer system of claim 13 wherein the control logic is configured to adjust the read pointer to cause a fill word to be read twice in response to detecting that the write pointer leads the read pointer by less than a predetermined minimum amount; and wherein the control logic is configured to adjust the write pointer to cause a fill word to be deleted in response to detecting that the write pointer leads the read pointer by more than a predetermined maximum amount.

15. The buffer system of claim 14 wherein the control logic is configured to act with a high priority to skip a fill word in response to detecting that the write pointer leads the read pointer by more than a high-priority threshold amount and to act with a low priority to skip a fill word in response to detecting that the write pointer leads the read pointer by more than the predetermined maximum amount, but less than the high-priority threshold amount.

16. The buffer system of claim 13 further comprising fill word logic configured to add a fill word bit to each word before the word is stored in the circular buffer and to set the fill word bit to indicate whether the word is a fill word.

17. The buffer system of claim 13 wherein the control logic is configured to compare the positions of the read and write pointers by subtracting the value of one of the pointers from the value of the other of the pointers.

18. The buffer system of claim 17 wherein the control logic is configured to compare the positions of the read and write pointers when one of the pointers has a value of 0.

19. The buffer system of claim 14 wherein the control logic is configured to determine an amount by which the write pointer leads the read pointer and to adjust the read pointer in response to the amount only if the amount is maintained for at least half of a clock cycle.

20. The buffer system of claim 13 wherein the control logic is configured to delay comparing the positions of the read and write pointers after adjusting one of the read and write pointers in response to the relative positions of the read and write pointers.

21. A device configured to be installed in a network having a loop topology, the device comprising:

an input port for receiving incoming data from a preceding device in the loop, wherein the incoming data is clocked by an external clock signal;

an output port for transmitting outgoing data to a subsequent device in the loop, wherein the outgoing data is clocked by an internal clock signal;

a circular buffer having a plurality of storage locations, wherein the circular buffer is coupled to the input and output ports and configured to store data in the plurality of storage locations;

write logic coupled to the input port and the circular buffer, wherein the write logic is configured to write received data to storage locations in the circular buffer indicated by a write pointer, wherein the write pointer is advanced after data is written to the indicated storage locations, and wherein data is written to the circular buffer at rate of the external clock signal;

read logic coupled to the output port and the circular buffer, wherein the read logic is configured to read data from the storage locations in the circular buffer indicated by a read pointer and transmit the read data to the output port, wherein the read pointer is advanced after data is read from the indicated storage locations, and wherein data is read from the circular buffer at rate of the internal clock signal; and

control logic configured to determine whether the read logic is reading data out of the circular buffer more quickly or more slowly than the write logic is writing data into the circular buffer, wherein if the read logic is reading data out of the circular buffer more quickly or more slowly than the write logic is writing data into the circular buffer, the control logic is configured to modify the position of at least one of the read and write pointers to compensate therefor.

22. The device of claim 21 wherein if the control logic determines that the read logic is reading data out of the circular buffer more quickly than the write logic is writing data into the circular buffer, the control logic is configured to position the read pointer to cause one of the storage locations in the circular buffer to be read two consecutive times.

23. The device of claim 22 wherein the storage location in the circular buffer which is read twice contains a fill word.

24. The device of claim 21 wherein if the control logic determines that the read logic is reading data out of the circular buffer more slowly than the write logic is writing data into the circular buffer, the control logic is configured to position the write pointer to overwrite one of the storage locations in the circular buffer one time.

25. The device of claim 24 wherein the storage location in the circular buffer which is skipped contains a fill word.

26. The device of claim 21 wherein the control logic is configured to determine whether the read logic is reading data out of the circular buffer more quickly or more slowly than the write logic is writing data into the circular buffer by comparing the positions of the read and write pointers.

27. The device of claim 26 wherein if the write pointer leads the read pointer by more than a predetermined maximum amount, the read logic is reading data out of the circular buffer more slowly than the write logic is writing data into the circular buffer and wherein if the write pointer leads the read pointer by less than a predetermined minimum amount, the read logic is reading data out of the circular buffer more quickly than the write logic is writing data into the circular buffer.

SYSTEM AND METHOD FOR JITTER COMPENSATION IN DATA TRANSFERS

ABSTRACT OF THE INVENTION

A system and method for compensating for differences between a recovered receive clock and an internal transmit clock in an elastic buffer and thereby preventing corruption of data. In one embodiment, the system comprises a circularly accessed buffer coupled to read and write logic. The read and write logic read and write to locations within the circular buffer as indicated by respective read and write pointers. The system further comprises control logic which compares the pointers to determine whether the buffer is approaching an underflow or overflow condition and adds or deletes fill words between frames of data to compensate for the underflow or overflow condition. In one embodiment, the system includes fill word logic which is configured to add a fill word bit to each received word and to set or clear the fill word bit to indicate whether or not the corresponding word is a fill word.

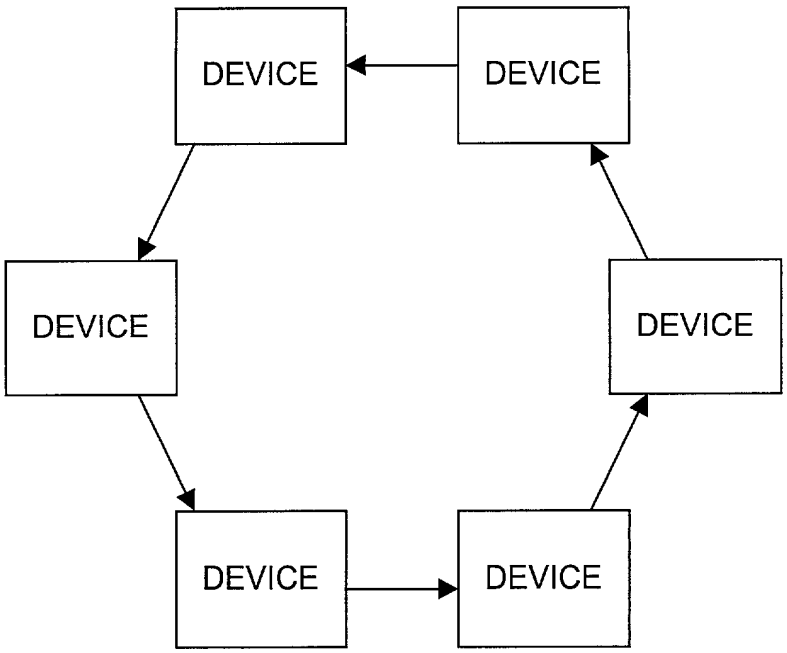

Fig. 1

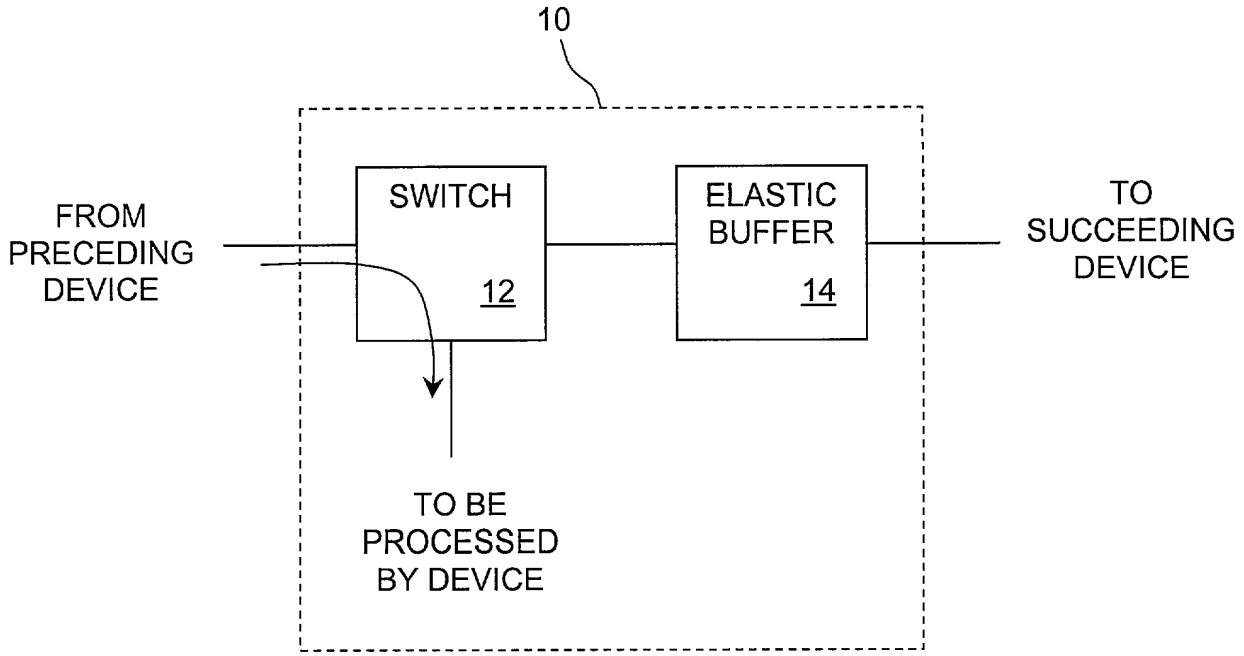
10
FROM
PRECEDING
DEVICE
SWITCH
12
ELASTIC
BUFFER
14
TO
SUCCEEDING
DEVICE
TO BE
PROCESSED
BY DEVICE

Fig. 2a

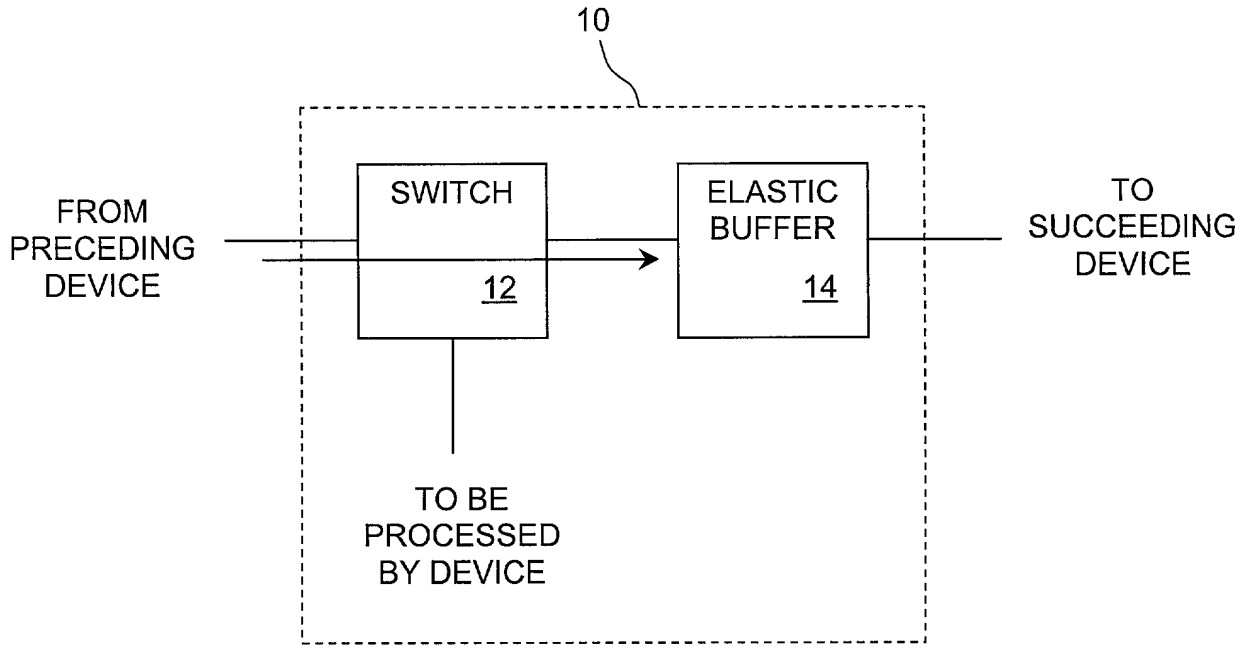
10
FROM
PRECEDING
DEVICE
SWITCH
12
ELASTIC
BUFFER
14
TO
SUCCEEDING
DEVICE
TO BE
PROCESSED
BY DEVICE

Fig. 2b

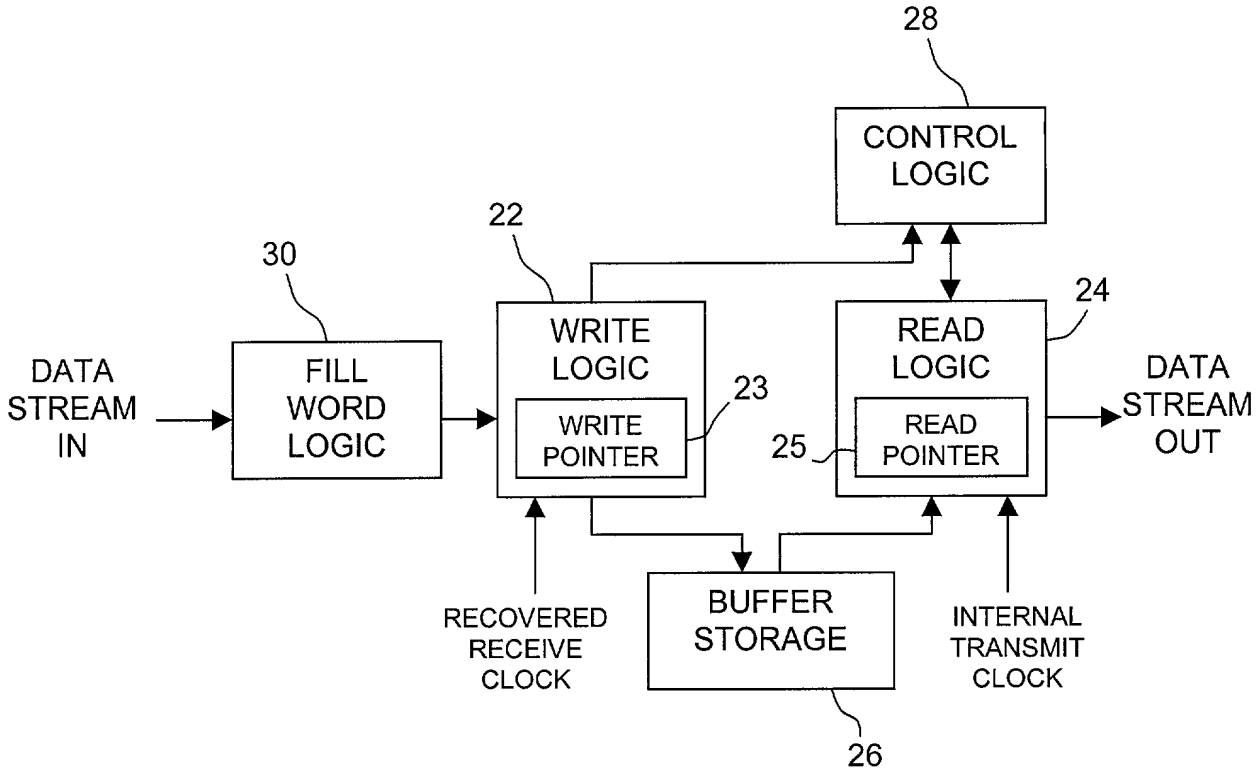
28
CONTROL LOGIC
22
30
24
WRITE LOGIC
READ LOGIC
DATA STREAM IN
FILL WORD LOGIC
23
WRITE POINTER
25
READ POINTER
DATA STREAM OUT
RECOVERED RECEIVE CLOCK
BUFFER STORAGE
INTERNAL TRANSMIT CLOCK
26

Fig. 3

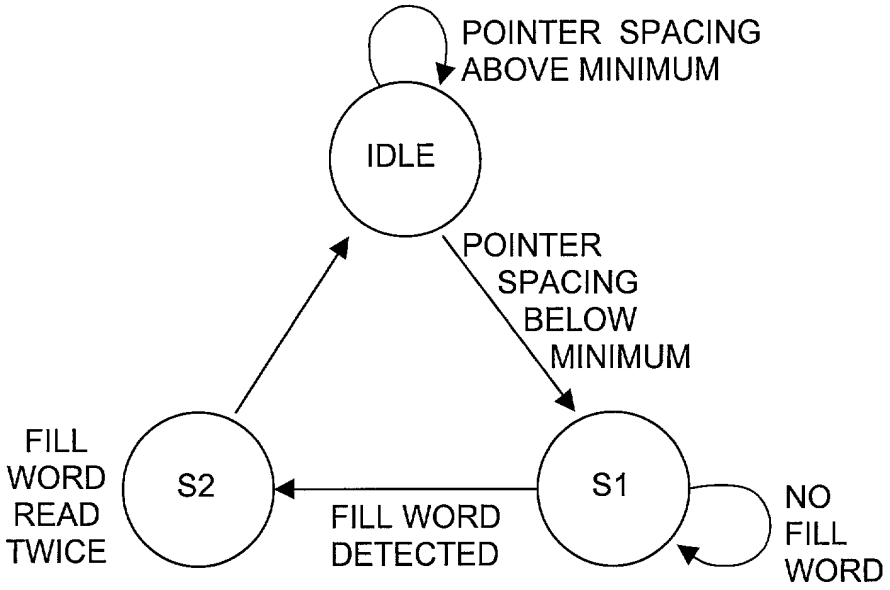
POINTER SPACING
ABOVE MINIMUM
IDLE
POINTER
SPACING
BELOW
MINIMUM
FILL
WORD
READ
TWICE
S2
S1
NO
FILL
WORD
FILL WORD
DETECTED

Fig. 4a

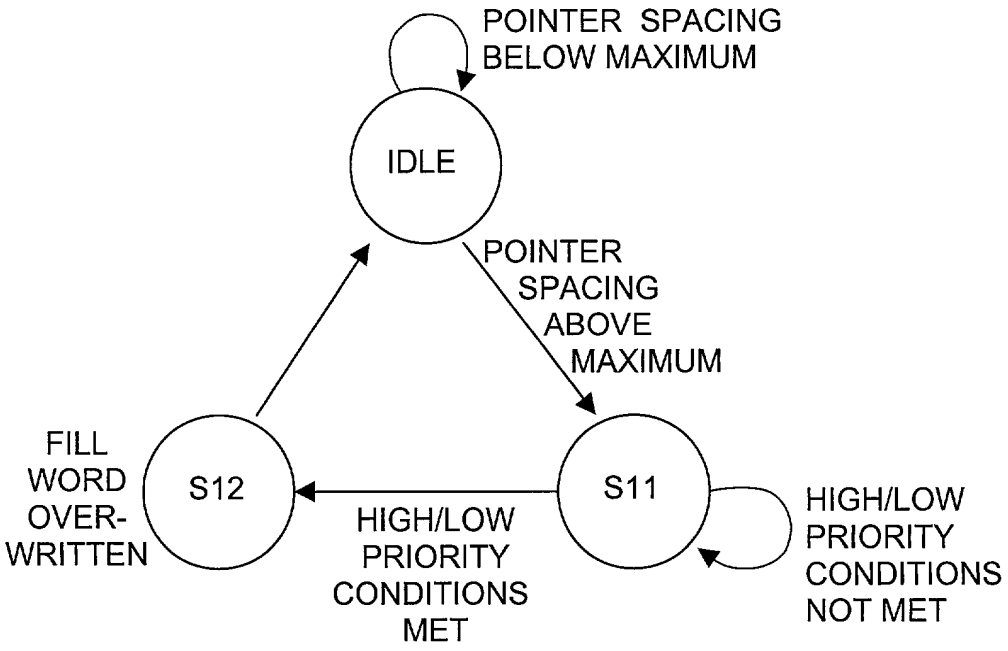
POINTER SPACING
BELOW MAXIMUM
IDLE
POINTER
SPACING
ABOVE
MAXIMUM
FILL
WORD
OVER-
WRITTEN
S12
S11
HIGH/LOW
PRIORITY
CONDITIONS
NOT MET
HIGH/LOW
PRIORITY
CONDITIONS
MET

Fig. 4b

Please type a plus sign (+) inside this box [+]

PTO/SB/01

DECLARATION FOR UTILITY OR DESIGN PATENT APPLICATION (37 CFR 1.63)

[X] Declaration Submitted with Initial Filing [ ] Declaration Submitted after Initial Filing

Attorney Docket No.	**CROSS1410-1**
First Named Inventor	**Michael A. Nelson, et al.**
COMPLETE IF KNOWN	
Filing Date	
Application Number	
Group Art Unit	
Examiner	

As a below named inventor, I hereby declare that:

My residence, post office address, and citizenship are as stated below to my name.

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled:

SYSTEM AND METHOD FOR JITTER COMPENSATION IN DATA TRANSFERS

(Title of Invention)

the specification of which was filed on (MM/DD/YYYY) [ ]

as United States Application Number of PCT International Application Number [ ]

and was amended on (MM/DD/YYYY) *(if applicable)* [ ]

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment specifically referred to above.

I hereby state I do not know and do not believe that said invention, design or discovery was ever known or used in the United States of America before my invention or discovery thereof, or patented or described in any printed publication in any country before my invention or discovery thereof, or more than one year prior to this application, or in public use or on sale in the United States of America more than one year prior to this application; that said invention, design or discovery has not been patented or made the subject of an inventor's certificate issued prior to the date of this application in any country foreign to the United States of America on an application filed by me or my legal representatives or assigns; and that I acknowledge the duty to disclose to the U.S. Patent and Trademark Office all information known to me which is material to the patentability as defined in 37 CFR 1.56.

I hereby claim foreign priority benefits under 35 U.S.C. 119(a)-(d) or 365(b) of any foreign application(s) for patent or inventor's certificate, or 365(a) of any PCT international application which designated at least one country other than the United States of America, listed below and have also identified below, by checking the box, any foreign application for patent or inventor's certificate, or of any PCT international application having a filing date before that of the application on which priority is claimed.

Prior Foreign Application Number(s)	Country	Foreign Filing Date (MM/DD/YYYY)	Priority Not Claimed	Certified Copy Attached? YES	NO

Additional foreign application numbers are listed on a supplemental priority data sheet PTO/SB/02B attached hereto:

I hereby claim the benefit under 35 U.S.C. 119(e) of any United States provisional application(s) listed below:

Application Number(s)	Filing Date (MM/DD/YYYY)	
60/202,720	05/08/00	[ ] Additional provisional application numbers are listed on a supplemental priority data sheet PTO/SB/02B attached hereto

DECLARATION -- Utility or Design Patent Application

I hereby claim the benefit under 35 U.S.C. 120 of any United States Application(s), or 365(c) of any PCT international application designating the United States of America, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States or PCT International application in the manner provided by the first paragraph of 35 U.S.C. 112, I acknowledge the duty to disclose information which is material to patentability as defined in 37 CFR 1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this application.

U.S. Parent Application or PCT Parent Number	Parent Filing Date (MM/DD/YYYY)	Parent Patent Number *(If applicable)*

Additional U.S. or PCT international application numbers are listed on a supplemental priority data sheet PTO/SB/02B attached hereto.

As a named inventor, I hereby appoint the following registered practitioner(s) associated with **Customer ID No. 25094** to prosecute this application and to transact all business in the Patent and Trademark Office connected therewith:

25094
PATENT _TRADEMARK OFFICE

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. 1001 and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Name of Sole or First Inventor:							
Given Name (first and middle [if any])				Family Name or Surname			
Michael A.				**Nelson**			
Inventor's Signature				Date	10/18/00		
Residence Address	**4213 Southwest Huber Street**			Citizenship	**United States**		
City	**Portland**	State	**OR**	Zip	**97219**	Country	**United States**
Post Office Address	**same**						

Name of Additional Inventor:							
Given Name (first and middle [if any])				Family Name or Surname			
Thomas W.				**Bucht**			
Inventor's Signature				Date	10/18/2000		
Post Office Address	**18655 Camelot Lane**			Citizenship	**United States**		
City	**Hillsboro**	State	**OR**	Zip	**97123**	Country	**United States**
Post Office Address	**same**						

Name of Additional Inventor:							
Given Name (first and middle [if any])				Family Name or Surname			
Inventor's Signature				Date			
Residence: City		State		County		Citizenship	
Post Office Address							
Post Office Address							
City		State		Zip		Country	

Direct all correspondence to Customer ID No. 25094:							
Name	**Mark L. Berrier**						
Address							
City				*State*		*Zip*	
Country		*Telephone*	**(512) 457-7016**		*Fax*	**(512) 457-7070**	

\AU\4044977.1
103671-991411